

Universidad de las Ciencias Informáticas



Universidad de las Ciencias Informáticas

Facultad 1

Análisis y diseño del evaluador de sitios web cubanos utilizando webviews

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autor:

Ernesto Miguel Aguila Caceres

Tutores:

Msc. Ariagna González Landeiro

Ing. Rubén Reynaldo Bonachea

La Habana, diciembre de 2021

“Año 63 de la Revolución”

DECLARACIÓN DE AUTORÍA

Yo **Ernesto Miguel Aguila Caceres**, con carné de identidad **97032708809**, declaro por este medio que soy el autor principal del trabajo titulado “Evaluador de sitios web cubanos” y autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Para que así conste firmamos la presente a los _____ días del mes de _____

Del año _____



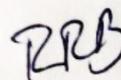
Ernesto Miguel Aguila Caceres

Autor



Msc. Ariagna González Landeiro

Tutor



Ing. Rubén Reynaldo Bonachea

Tutor

Resumen

En la actualidad existe demasiada información digital al alcance de todos, ese exceso de información hace que a los usuarios se les dificulte encontrar lo que buscan en determinados momentos. Como parte de ese mundo digital se encuentran los sitios web cubanos, los cuales son evaluados en la Universidad de Ciencias Informáticas por parte del equipo del Monitor Red Cuba de forma continua para determinar fallos y así contribuir a mejorar su posicionamiento en la red. La presente investigación permitió desarrollar una herramienta informática implementada con tecnologías webviews dedicada a evaluar dichos sitios para así contribuir al trabajo realizado por la Dirección de Proyectos Especiales y al mejoramiento de la informatización del país. Para la implementación de la propuesta de solución, guiado por la metodología XP se seleccionaron como principales tecnologías: el marco de trabajo Springs 2.6, para la programación en JAVA 16.0.1 y Visual Paradigm 8.0 como herramienta para el modelado.

Palabras Claves: posicionamiento, sitios web, herramienta informática, webviews

Índice

Introducción	6
Capítulo 1. Fundamentos Teóricos	9
1.1 Introducción.....	9
1.2 Principales Conceptos.....	9
1.3 Estudio de sistemas homólogos	9
1.4 Descripción de las herramientas y Tecnologías.....	13
1.4.1 Lenguaje de Programación	13
1.4.2 Lenguaje de Modelado.....	13
1.4.3 Framework.....	14
1.4.4 Entorno de Desarrollo	14
1.4.4 Herramienta CASE	15
1.4.5 Gestor de Bases de Datos	15
1.4.6 Colas de mensajes	15
1.5 Metodología de Software.....	16
1.6 Conclusiones.....	16
Capítulo 2. Diseño de la Propuesta de Solución	17
2.1 Introducción.....	17
2.2 Propuesta del sistema	17
2.3 Modelo Conceptual	18
2.4 Especificación de los Requisitos del Software	18
2.4.1 Requisitos funcionales	18
2.4.2 Requisitos no funcionales	19
2.5 Modelo de Casos de Uso del Sistema.....	19
2.5.1 Diagrama de Caso de Uso del Sistema.....	19
2.7 Historias de Usuario	20
2.8 Planificación	20
2.8.1 Plan de Iteraciones	20

2.8.2 Plan de Entrega	21
2.9 Diagrama de Clase.....	21
2.10 Modelo Entidad-Relación	22
2.11 Modelo de Despliegue.....	23
2.12 Arquitectura del Sistema	23
Capítulo 3 Pruebas del sistema	25
Bibliografía.....	28
Anexos	30

Introducción

El uso cada vez más amplio de Internet es palpable en la actualidad, elemento que ha sido evidenciado por las diferentes compañías y herramientas que estudian este fenómeno en el mundo. Por solo citar algún ejemplo según el estudio anual del 2019 presentado por HootSuite (plataforma para la gestión de redes sociales) el 56% de la población cubana tiene acceso a Internet. Lo cual ha sido posible entre otros aspectos por el aumento del uso de dispositivos móviles inteligente, pues según datos de la misma herramienta más de los 6,47 millones de usuarios acceden desde sus teléfonos móviles (González Landeiro, Castillo Galán, & Valdés García, 2020).

A lo largo de los años ha ocurrido un incremento en la generación de contenido y servicios, lo cual dificulta cada vez más la tarea de los desarrolladores de dar a conocer sus creaciones, por lo que se deben idear formas de hacer llegar esa información a las masas.

Utilizar la vía de la voz popular para dar a conocer la existencia de algún contenido digital es una de las vías, pero no es la más efectiva, ni la que debe perdurar en el tiempo. Sin embargo, una parte cada vez mayor del tráfico que se genera hacia las plataformas digitales, llega a través de la página de resultados de los buscadores y de las redes sociales, en lugar de llegar directamente como resultado de la navegación en su propia página web (González Landeiro, Castillo Galán, & Valdés García, 2020).

Lograr que los contenidos se muestren entre las primeras opciones es un reto diario para los profesionales dedicados a esta tarea, los cuales se enfocan en lograr que sus sitios tengan posiciones destacadas entre los resultados de búsquedas en la web, de ahí que se decidiera desarrollar un Monitor de Sitios Web Cubanos, el cual se encargará de medir y evaluar variables relacionadas con el rendimiento de los sitios web para así mejorar su posicionamiento.

Problema de la investigación

¿Cómo mejorar la capacidad del Monitor de Sitios web de evaluar un rango mayor de variables?

Objeto de estudio

Objeto de estudio: Procesos de análisis de las técnicas SEO enfocadas al posicionamiento web.

Campo de Acción: Monitor de Sitios web cubano.

Objetivo General

Desarrollar el análisis y diseño del evaluador de variables del monitor de sitios webs cubanos utilizando tecnologías webviews.

Objetivos específicos

- Establecer los fundamentos teórico-metodológicos para la construcción del

marco teórico-conceptual a utilizar en la presente investigación.

- Estudiar las tendencias de la evaluación de los indicadores SEO en herramientas informáticas con estos fines.
- Elaborar el análisis y diseño para la herramienta informática a desarrollar.

Variable Independiente: sistema informático.

Variable Dependiente: evaluación de sitios web cubanos.

Hipótesis

Con la elaboración de un análisis del funcionamiento del evaluador del monitor de sitios webs cubanos se podría llegar a entender como mejorar la capacidad del mismo de evaluar un mayor rango de variables.

Tareas de Investigación

1. Elaborar el perfil de trabajo de diploma.
2. Escribir el marco metodológico de la tesis.
3. Realizar estudio sobre herramientas similares.
4. Escribir el marco teórico referencial de la tesis.
5. Escribir los resultados de la investigación.
6. Presentar la investigación.

Métodos Científicos

Para darle cumplimiento a las tareas de investigación, se utilizaron los siguientes métodos de investigación:

Teóricos

- Analítico – Sintético: Empleado para el análisis de los elementos esenciales referentes a las teorías, documentos y literatura en general relacionada con SEO.
- Modelación: Utilizado en la representación, mediante el uso de diagramas, de las características del sistema, relaciones entre objetos; y las actividades que intervienen en el proceso de desarrollo de la herramienta informática para el análisis SEO.
- Histórico-Lógico: Empleado en la recopilación de información acerca de diversos conceptos, tecnologías y sistemas homólogos.

Empíricos

- Entrevista: Empleado para definir la metodología y entender el funcionamiento del negocio y la herramienta.

Estructura del documento

Este trabajo de diploma tendrá la estructura siguiente: Introducción, 3 Capítulos, Conclusiones y Recomendaciones.

Capítulo 1. Fundamentos Teóricos, análisis de conceptos relacionados con el servicio de posicionamiento web más conocido por *Search Engine Optimization* (SEO).

Capítulo 2. Análisis y diseño de una propuesta de solución.

Capítulo 3. Propuesta de Pruebas

Posibles Resultados

1. Investigación que contribuya a entender cómo mejorar la capacidad de análisis de variables del evaluador de sitios web cubanos.

Capítulo 1. Fundamentos Teóricos

1.1 Introducción

Con el desarrollo de este capítulo se busca lograr una comprensión por parte del lector acerca de los conceptos básicos relacionados con la presente investigación. Para lograrlo se describirán conceptos relacionados con el tema y se describirán las bases para la propuesta. Finalizando el capítulo se establecerá la metodología a usar, así como los conceptos de las tecnologías que darán forma al proyecto.

1.2 Principales Conceptos

Mediante el empleo del método histórico-lógico se realizó un análisis de conceptos relacionados con la temática de posicionamiento web para así definir cuáles serían de utilidad para la investigación. Luego del análisis se determinaron los siguientes conceptos:

SEO: Acrónimo de *Search Engine Optimization* -en castellano optimización de motores de búsqueda-, son el conjunto de acciones y técnicas que se emplean para mejorar el posicionamiento en buscadores de un sitio web en Internet, dentro de los resultados orgánicos en los motores de búsqueda como, Google, Bing o Yahoo [2].

Factores de Posicionamiento: Existen más de 200 factores de posicionamiento que, como su propio nombre indica, hacen que la página aparezca en una posición u otra en las páginas de resultados de Google. Estos factores están en continuo cambio y se dividen en 2 tipos: Factores internos (on-site) y factores externos (off-site) [3].

Factores Externos: Son aquellos que no se pueden controlar directamente. Señalan a los buscadores la autoridad que tiene una web, es decir, si es de confianza o no. Nadie sabe exactamente como se establece con exactitud la autoridad de una web, aunque es probable que existan numerosos factores que la definan. El comportamiento del usuario, la calidad de los links externos y las señales sociales son algunos de los más importantes [3].

Factores Internos: Son aquellos que son relativos a una web, es decir, aquellos que se pueden controlar directamente. Su objetivo principal es hacer la vida más fácil a los buscadores cuando acceden a las páginas. La calidad del contenido, la arquitectura web y el código HTML son los factores más relevantes de una web [3].

1.3 Estudio de sistemas homólogos

Google Analytics

La principal función de esta herramienta es medir el impacto de tu sitio web y el comportamiento de sus usuarios. La finalidad es entender el rendimiento del proyecto y cómo optimizarlo [4].

Google Analytics puede realizar un seguimiento de prácticamente todas las interacciones realizadas en las páginas, con un alto nivel de detalle. Un error común es querer medirlo todo. Por esto, la planificación estratégica del seguimiento, medición y análisis es esencial para poder usar bien las funciones de esta herramienta [4].

Específicamente, lo que podrás lograr con Google Analytics es:

- Conocer a tus usuarios
- Explorar el comportamiento
- Medir las interacciones
- Analizar el rendimiento de acciones

Conoce a tus usuarios

Encuentra información sobre quiénes visitan tu sitio web. La herramienta te mostrará desde qué país y con qué dispositivos se conectan, qué idioma usan, su género y edad, entre otros (Grill Taylor-Made Performance, 2021).

Además, podrás ver sus intereses y afinidades para desarrollar campañas personalizadas. Esta información también puede ser relevante para la creación de buyer personas o arquetipos de usuario [4].

Explorar el comportamiento

Estudia cómo y cuándo los usuarios navegan por tu sitio web. Podrás ver detalles sobre qué contenidos consumen, cuánto tiempo se quedan en el sitio, cuántas páginas visitan y dónde abandonan el sitio, entre otros [4].

También existe la posibilidad de ver en tiempo real cuántas personas están navegando los contenidos, lo que puede ser muy útil para monitorear el efecto inmediato de una campaña [4].

Medir las interacciones

Con la correcta configuración, podrás ver en qué botones hacen click los usuarios para ver videos, suscribirse a listas, solicitar información o descargar contenidos. De esta forma podrás ver qué llamados están funcionando y entender mejor los intereses de los usuarios [4].

La medición de interacciones también puede alertarte sobre problemas técnicos o de diseño en tu sitio web. Por ejemplo, si una página ofrece un e-book gratuito y recibe muchas visitas, pero nadie lo descarga, puede ser que exista un error que esté impidiendo que las personas completen la acción [4].

Analizar el rendimiento de canales

La herramienta permite segmentar los datos para analizar a las visitas que vienen desde un canal específico. Los posibles canales son redes sociales, email, buscadores como Google y Bing, anuncios de pago, otros sitios web y tráfico directo. Este último se refiere a las personas que llegaron al sitio web escribiendo la URL directamente en el navegador o usando un marcador [4].

Al analizar a las visitas que provienen de un canal específico podrás entender la efectividad de ese canal. Por ejemplo, si todos los usuarios que llegan por anuncios de pago se van inmediatamente del sitio, necesitas mejorar tus campañas. En este caso, es posible que los anuncios estén prometiendo algo diferente a lo que el sitio web ofrece [4].

Woorank

El mismo sitio de la herramienta de análisis SEO Woorank se define como una “aplicación con la que puedes generar informes de sitios web, en los que se incluyen datos significativos sobre su estado, así como una lista de consejos, recomendaciones a implementar con tal de optimizar la presencia online de dichos sitios” [5].

Es una herramienta que te ayuda a realizar un seguimiento de todo lo que afecta el rendimiento SEO de tu sitio web y descubrir cualquier problema que no hayas notado antes [6].

Para esto deberás escribir la URL de tu sitio web y te presentará una auditoría detallada de SEO para todos los factores de clasificación importantes [6].

Te preguntará dónde está el milagro. En el ahorro del tiempo, porque si intentas hacer este proceso manualmente, te llevará muchas horas, sin embargo, si usas WooRank, obtendrás el informe en solo unos segundos [6].

Una vez que obtienes esta auditoría SEO, puedes usar la información para corregir cualquier error que pueda dañar tu clasificación [6].

Además, puedes conectar tu cuenta a Google Search Console, Google Analytics y tu página de Facebook, para obtener los mejores resultados posibles mientras analizas cada página de tu sitio por separado o rastreas tus palabras clave [6].

En pocas palabras, esta es una poderosa herramienta que puede ayudarte a obtener el máximo de SEO para cualquier proyecto [6].

GTMetrix

GTmetrix es una herramienta que permite medir la velocidad de una página web y, también, otros aspectos que impactan en el rendimiento de un sitio web, en el tiempo de carga y, por ende, en la satisfacción de un usuario [7].

Desde el año 2009, a través de su plataforma web, tienen disponible la función para hacer el test de velocidad gratis. Asimismo, han añadido otras funcionalidades, como los reportes, la video simulación, la posibilidad de replicar cargas desde distintas locaciones y más información [7].

Entre los aspectos que reportan, están los siguientes [7]:

- Tiempo de carga de contenido pesado.
- Segundos para que el usuario pueda interactuar.
- Carga de las imágenes.
- Tiempo que se gasta conectando con el servidor.
- El uso de código Javascript y CSS.
- Tiempo de espera por parte del usuario.

Semrush

Se trata de una herramienta de las que llamamos “todo en 1”, sobre la que va a girar tu estrategia SEO. Una de esas que te facilita la vida porque te ahorra tener que descargar o instalar en el navegador diferentes programas o extensiones para cuestiones como hacer seguimiento de posiciones, analizar los errores OnPage, analizar las consultas de búsqueda de los usuarios (keywords) o revisar los enlaces entrantes de una determinada web [8].

Esta es una de las funcionalidades más interesantes y donde se marca la diferencia. Ya que no todas las herramientas SEO tienen acceso a las mismas bases de datos de consultas. Con esta función podrás analizar la cantidad de búsquedas mensuales (Volumen) que tiene un término en un determinado buscador (Google.es, Google.de, Google.co.uk...) y comprobar si es rentable o no apostar por él (...) [8].

Luego de un análisis de diferentes herramientas destinadas a la evaluación de sitios web se obtuvo como resultado que es necesario ampliar el espectro de variables analizadas debido a que a pesar de que el funcionamiento es parecido, quedan fuera del proceso algunas de gran impacto en el posicionamiento web, lo que causa que el análisis brindado por el evaluador no de un informe tan completo como otras aplicaciones.

1.4 Descripción de las herramientas y Tecnologías

1.4.1 Lenguaje de Programación

En informática se conoce como lenguaje de programación a un programa destinado a la construcción de otros programas informáticos. Su nombre debe a que comprende un lenguaje formal que está diseñado para organizar algoritmos y procesos lógicos que serán luego llevados a cabo por un ordenador o sistema informático, permitiendo controlar así su comportamiento físico, lógico y su comunicación con el usuario humano [9].

Java

El lenguaje de programación Java fue desarrollado originalmente por James Gosling, de *Sun Microsystems* (constituida en 1983 y posteriormente adquirida el 27 de enero de 2010 por la compañía Oracle), y publicado en 1995 como un componente fundamental de la plataforma Java de *Sun Microsystems*. Su sintaxis deriva en gran medida de C y C++, pero tiene menos utilidades de bajo nivel que cualquiera de ellos. Las aplicaciones de Java son compiladas a bytecode (clase Java), que puede ejecutarse en cualquier máquina virtual Java (JVM) sin importar la arquitectura de la computadora subyacente [10].

El lenguaje Java se creó con cinco objetivos principales:

1. Debería usar el paradigma de la programación orientada a objetos.
2. Debería permitir la ejecución de un mismo programa en múltiples sistemas operativos.
3. Debería incluir por defecto soporte para trabajo en red.
4. Debería diseñarse para ejecutar código en sistemas remotos de forma segura.
5. Debería ser fácil de usar y tomar lo mejor de otros lenguajes orientados a objetos, como C++ [10].

1.4.2 Lenguaje de Modelado

Una imagen vale más que mil palabras. Es por eso que se creó la generación de diagramas con el Lenguaje Unificado de Modelado (UML): para forjar un lenguaje visual común en el complejo mundo del desarrollo de software que también fuera comprensible por los usuarios de negocios y quienquiera que desee entender un sistema [11].

El Lenguaje Unificado de Modelado (UML) fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento. UML tiene aplicaciones más allá del desarrollo de software, p. ej., en el flujo de procesos en la fabricación [11].

1.4.3 Framework

En el desarrollo de software, un framework es una estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software. Típicamente puede incluir soportes de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto [12].

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio, y provee una estructura y una especial metodología de trabajo, la cual extiende o utiliza las aplicaciones del dominio [12].

Spring Framework

Spring es un framework que nos permite desarrollar aplicaciones con el Lenguaje de Programación Java. Spring Framework cuenta con muchas características que hacen a los desarrolladores más productivos [13].

Spring Boot

Es una herramienta que nos permite crear un proyecto con Spring Framework, solo que Spring Boot elimina ciertas configuraciones repetitivas requeridas para desplegar la aplicación [13].

Entre las características de la herramienta Spring Boot se encuentran:

- Permite hacer una configuración automática para que Spring Framework funcione.
- Ofrece un servidor incorporado para evitar la complejidad en el despliegue.
- Cuenta con dependencias iniciales que simplifican la compilación y configuración de una aplicación Spring [13].

1.4.4 Entorno de Desarrollo

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE1 es un producto libre y gratuito sin restricciones de uso [14].

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las API de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software [14].

1.4.4 Herramienta CASE

Visual Paradigm es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación [15].

Visual Paradigm presenta todos los diagramas UML y herramientas ERD esencialmente en el diseño de sistemas y bases de datos. Las innovadoras herramientas de modelado como Resource Catalog, Transitor y Nicknamer hacen que el modelado de sistemas sea fácil y rentable. Doc. Composer le permite producir especificaciones de diseño detalladas listas para usar en discusión con solo unos pocos clics [16].

1.4.5 Gestor de Bases de Datos

MySQL es un sistema de gestión de base de datos relacional desarrollado bajo licencia dual: Licencia publica general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo [17].

Es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido [17].

1.4.6 Colas de mensajes

RabbitMQ es un bróker de mensajería de código abierto, distribuido y escalable, que sirve como intermediario para la comunicación entre productores y consumidores [18].

Implementa el protocolo de mensajería de capa de aplicación *Advanced Message Queuing Protocol (AMQP)*, el cual está enfocado en la comunicación de mensajes asíncronos con garantía de entrega, a través de confirmaciones de recepción de mensajería desde el bróker al productor y desde los consumidores al broker [18].

De forma simplificada, en RabbitMQ se definen colas que van a almacenar los mensajes que envían los productores hasta que las aplicaciones consumidoras obtienen el mensaje y lo procesan. Esto permite diseñar e implementar sistemas distribuidos, en los cuales un sistema se divide en módulos independientes que se comunican entre si a través de mensajes [18].

1.5 Metodología de Software

Luego de una entrevista al equipo de desarrollo de la Dirección de Proyectos Especiales se determinó que en sus proyectos la metodología desarrollo de software empleada es XP, por lo cual esa será la metodología a utilizar en esta investigación.

1.6 Conclusiones

- Como conclusión se puede decir que es necesario tener bien claros los conceptos básicos de la problemática en cuestión.
- Analizar otros trabajos para comparar puntos fuertes y débiles que se puedan mejorar.
- Se decide utilizar como framework Spring Boot.
- Como metodología de desarrollo se empleará XP.

Capítulo 2. Diseño de la Propuesta de Solución

2.1 Introducción

En este capítulo se abordarán los aspectos fundamentales relacionados con el diseño del sistema a desarrollar. Como vía para definir las futuras funcionalidades de la aplicación y qué usuarios podrán tener acceso a las mismas, se generaron los artefactos relacionados a la especificación de los requerimientos funcionales y no funcionales que deberá poseer el software; así como la especificación de los casos de uso del sistema. En el presente capítulo se presenta los usuarios con sus responsabilidades y la captura de requisitos con la elaboración de las historias de usuario. Además, se presentan los conceptos fundamentales en un diagrama conceptual del negocio, un diagrama de clases y el modelo entidad-relación.

2.2 Propuesta del sistema

Como parte de entrevistas realizadas al equipo de desarrollo del Monitor de sitios web se obtuvo como resultado que el evaluador actual procesa un grupo de variables de gran impacto para el posicionamiento web, relacionadas con la usabilidad, la seguridad y la comunicación con la base de datos. La tecnología con la que se encuentra desarrollado le permite cargar el HTML de un sitio para luego extraer la información necesaria. Sin embargo, actualmente muchos sitios emplean contenido dinámico mediante CSS o Javascript, lo que ocasiona que el sistema a la hora de extraer el código, lo pase por alto. Entre esas variables que no son analizadas se encuentran el rendimiento del sitio, el peso total, entre otras, que hoy en día son de gran importancia para lograr un posicionamiento de calidad en la web, dando como consecuencia que la evaluación sea incompleta.

Se propone como solución el desarrollo del evaluador utilizando tecnologías webviews, la cual permitiría sincronizar el navegador con la aplicación y de esta forma cargar completamente el sitio permitiendo evaluar todo el contenido del mismo. Esta nueva implementación también dotará al evaluador de poder analizar en un futuro otro grupo de variables relacionadas con rendimiento, peso total y tiempo de carga real.

2.3 Modelo Conceptual

El proceso a modelar es el funcionamiento del evaluador de sitios webs, este comienza cuando el sistema accede al directorio de sitios web a través de una API REST para crear un listado de todos los sitios disponibles. Una vez obtenidos se encolan mediante RabbitMQ para que el proceso de análisis sea más organizado y así evitar un colapso. A continuación, se organizan los procesos de obtención de variables para luego proceder a analizarlas. Una vez que ya se tiene toda la información, se procede a guardarla en una base de datos mediante MySQL y en otra mediante InfluxDB, esta última para guardar las estadísticas de los resultados obtenidos, los cuales serán mostrados más adelante.

Figura 1: Modelo del Negocio

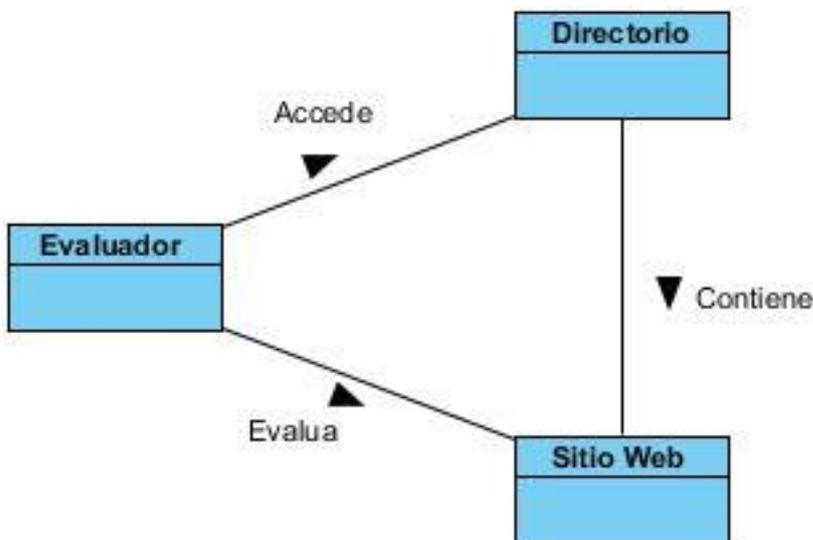


Figura 1: Modelo del Negocio

2.4 Especificación de los Requisitos del Software

2.4.1 Requisitos funcionales

- RF1. Desarrollar la librería que obtiene los datos del sitio web utilizando webviews.
- RF2. Adaptar los procesos de obtención de contenido.
- RF3. Descargar el contenido HTML de los sitios.
- RF4. Adaptar el sistema a la nueva forma de obtención de contenidos.
- RF5. Obtener los logs.
- RF6. Guardar los logs en la base de datos.
- RF7. Integración de Chrome al sistema con webviews.

2.4.2 Requisitos no funcionales

Software

Se debe poder instalar en el sistema operativo Ubuntu LTS.

Debe correr en la máquina virtual de java 11.

Usabilidad (No son requeridos)

Seguridad (No son requeridos)

Hardware (No son requeridos)

2.5 Modelo de Casos de Uso del Sistema

El modelo de casos de uso del sistema es un diagrama que explica de forma simbólica la interacción entre los actores y los casos de uso (CU) que representan las funcionalidades de la aplicación. Además, son representadas las relaciones existentes entre casos de uso. El actor es una entidad externa del sistema que participa en la ejecución del caso de uso. Generalmente el actor estimula el sistema con eventos de entrada o recibe algo de él.

2.5.1 Diagrama de Caso de Uso del Sistema

Figura 2: Casos de uso del sistema

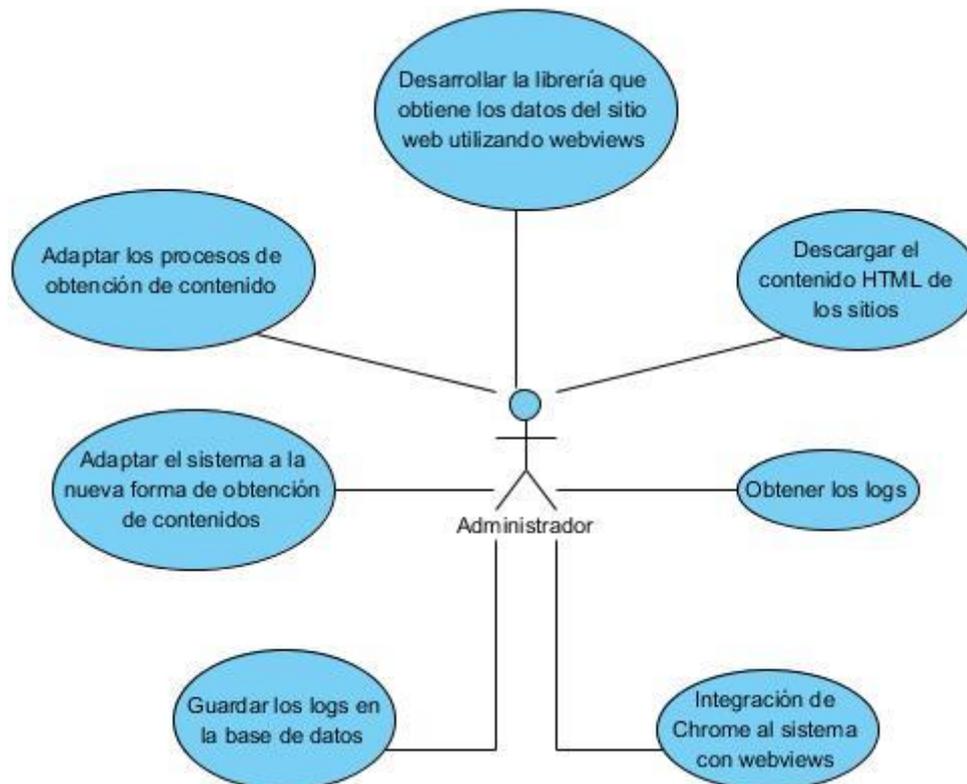


Figura 2: Casos de uso del sistema

2.7 Historias de Usuario

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento historias de usuario pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas [18].

A continuación, se muestra la historia de usuario: Desarrollar la librería que obtiene los datos del sitio web utilizando webviews.

Tabla 2: HU1- Desarrollar la librería que obtiene los datos del sitio web utilizando webviews

Historia de Usuario	
Número: 1	Nombre: Desarrollar la librería que obtiene los datos del sitio web utilizando webviews
Usuarios: Administrador	
Prioridad en el negocio: Alta	Nivel de complejidad: Alta
Estimación: 4 días	Iteración Asignada: 1
Descripción: El sistema emplea la nueva librería para obtener los datos de los sitios empleando tecnologías webviews.	
Observaciones: -	

Tabla 2: HU1- Desarrollar la librería que obtiene los datos del sitio web utilizando webviews

Para más Historias de usuario ver: [Anexos](#)

2.8 Planificación

2.8.1 Plan de Iteraciones

Todo proyecto que se guía por la metodología XP, se planifica en iteraciones de aproximadamente tres semanas de duración. Para cada iteración se define un conjunto de HU que se van a implementar [19].

Tabla 9: Plan de Iteraciones.

Iteraciones	Orden de las HU a Implementar	Tiempo de trabajo
1	- Desarrollar la librería que obtiene los datos del sitio web utilizando webviews - Adaptar los procesos de obtención de contenido	8
2	- Descargar el contenido HTML de los sitios. - Adaptar el sistema a la nueva forma de obtención de contenidos	8
3	- Obtener los logs. - Guardar los logs en la base de datos. - Integración de Chrome al sistema con webviews.	12

2.8.2 Plan de Entrega

El plan de entregas es el compromiso final del equipo de trabajo con los clientes. Esta es una cuestión de vital importancia para ambas partes, ya que la entrega tardía o temprana de la solución, repercute notablemente en la economía y la moral de los implicados [20].

Tabla 10: Plan de Entrega.

Plan de Entrega		
Iteración 1	Iteración 2	Iteración 3
1 de septiembre de 2021	9 de septiembre de 2021	17 de septiembre de 2021

2.9 Diagrama de Clase

El diagrama de clases describe gráficamente las especificaciones de las clases de software en una aplicación. A diferencia del modelo conceptual, un diagrama de este tipo contiene las definiciones de las entidades del software en vez de conceptos del mundo real. El UML no define concretamente un elemento denominado "diagrama clases del diseño", sino que se sirve de un término más genérico: "diagrama de clases". Se trata de una perspectiva desde el punto de vista del diseño de las entidades de software y no de una concepción analítica sobre los conceptos del dominio [21].

Figura 3: Diagrama de clase

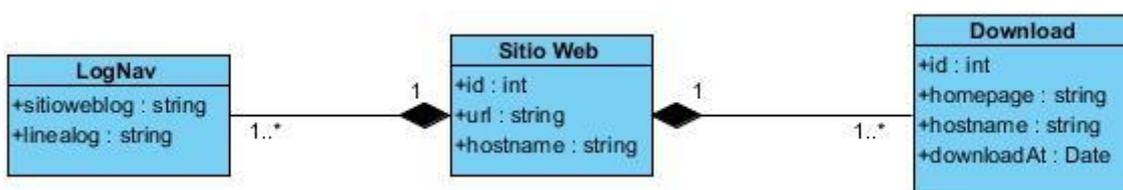


Figura 3: Diagrama de clase

2.10 Modelo Entidad-Relación

El Modelo Entidad Relación (ER) permite desarrollar un diseño de base de datos en un esquema de alto nivel conceptual sin considerar los problemas de bajo nivel como la eficiencia, el modelo implícito del administrador de base de datos o las estructuras físicas de los datos. El Modelo Entidad Relación se hizo muy popular para el diseño de base de datos y es usado extensivamente. Para aumentar su poder de expresión, muchos investigadores han introducido o propuesto ciertas extensiones a este modelo. Algunas de estas extensiones son importantes, mientras que otras agregan poco poder de expresión, pero proveen características auxiliares. Puesto que el Modelo Entidad Relación es ampliamente usado, es importante conocer qué extensiones han sido propuestas para este modelo y qué ofrecen estas extensiones a los usuarios [22].

Figura 4: Diagrama Entidad Relación

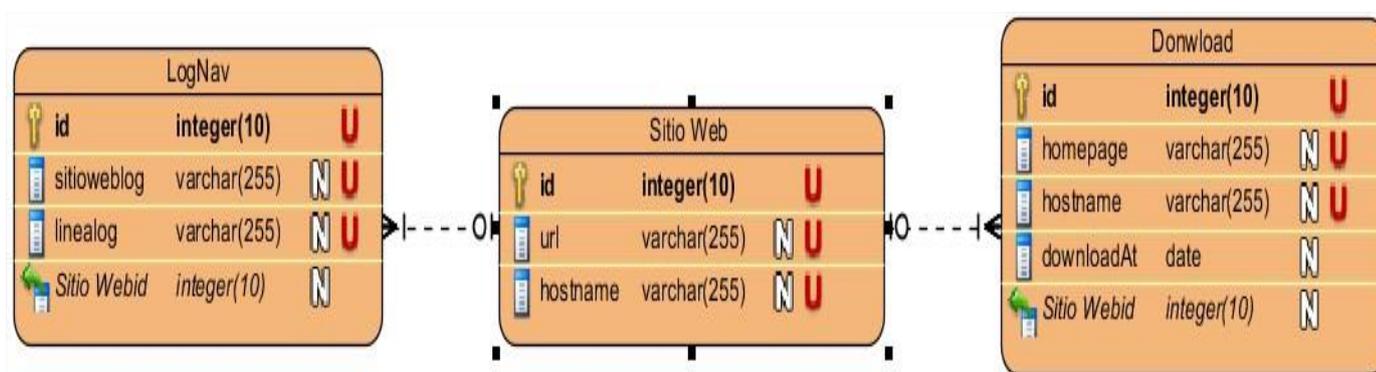


Figura 4: Diagrama Entidad Relación

2.11 Modelo de Despliegue

Diagrama de Despliegue es un tipo de diagrama del Lenguaje Unificado de Modelado que muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos [23].

Los diagramas de despliegue son los complementos de los diagramas de componentes que, unidos, proveen la vista de implementación del sistema. Describen la topología del sistema la estructura de los elementos de hardware y el software que ejecuta cada uno de ellos. Los diagramas de despliegue representan a los nodos y sus relaciones. Los nodos son conectados por asociaciones de comunicación tales como enlaces de red, conexiones TCP/IP [23].

Figura 5: Diagrama de Despliegue

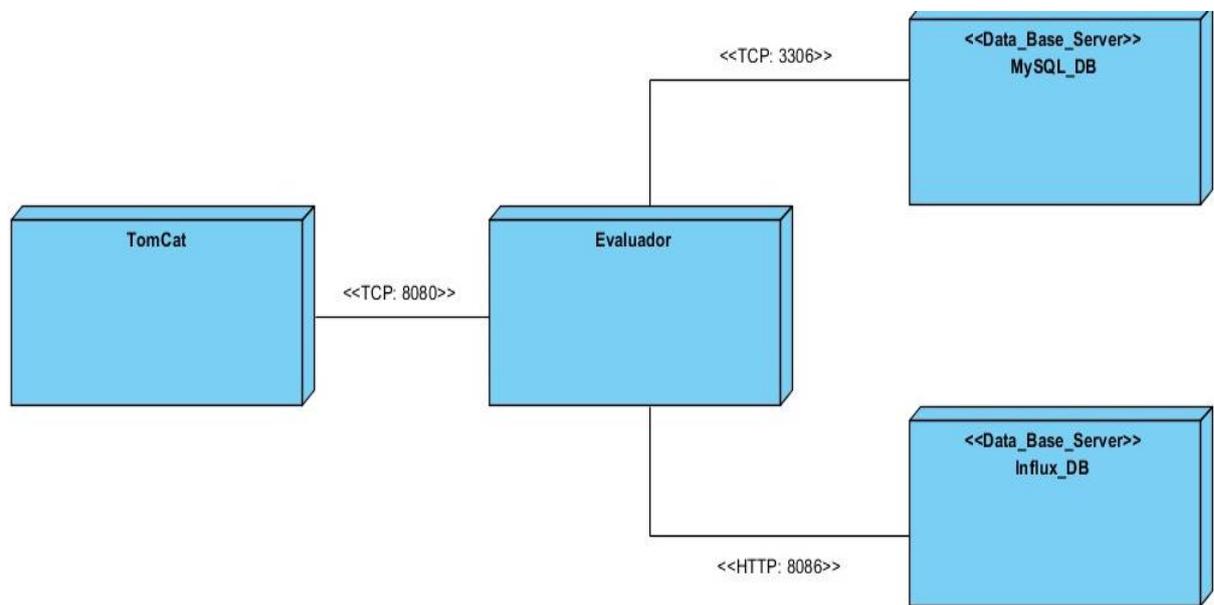


Figura 5: Diagrama de Despliegue

2.12 Arquitectura del Sistema

Modelo Vista Controlador (MVC). Es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El estilo de llamada y retorno MVC, se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista [24].

Descripción

- El Modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el Modelo y sus Vistas, y notificar a las Vistas cuando cambia el Modelo.
- La Vista es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario. Interactúa con el Modelo a través de una referencia al propio Modelo.
- El Controlador es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del Modelo o por alteraciones de la Vista. Interactúa con el Modelo a través de una referencia al propio Modelo.

Conclusiones

- Se ha diseñado un sistema específicamente para las necesidades empresariales concretas que se puede adoptar a través de la programación de software personalizado para adaptarse a las necesidades operativas de una organización.
- Se levantaron los requisitos funcionales y no funcionales.
- Se confeccionaron también las Historias de Usuarios
- Se obtuvo el Modelo Conceptual como apoyo para una mejor comprensión del negocio.

Capítulo 3 Propuesta de pruebas

Una parte importante del desarrollo de software es la etapa de pruebas, por lo que se decidió realizar un capítulo dirigido a propuestas de pruebas destinadas a comprobar la calidad y funcionamiento de la herramienta una vez que sea implementada. El software que se construye debe ser sometido a determinadas pruebas que corroboren la correspondencia entre el producto y los requisitos definidos en las etapas anteriores.

Una de las pruebas propuestas son los test unitarios, estos se realizan para comprobar el código y asegurarse que cada método funciona correctamente teniendo correspondencia con los planteado en los requisitos funcionales.

Conclusiones Parciales

Luego de las pruebas aplicadas al código se puede concluir que:

- Los métodos devuelven los resultados esperados.
- El sistema se comporta de manera correcta.

Conclusiones Generales

Con el desarrollo del presente documento se pudo arribar a las siguientes conclusiones:

- La elaboración del marco teórico referencial es de suma importancia para cualquier trabajo escrito ya que permite realizar el análisis teórico del problema, amplía el horizonte del estudio y guía al investigador para que se centre en su problema evitando desviaciones del planteamiento original.
- La definición de la metodología de desarrollo de software, definió las pautas para el proceso ordenado y detallado de la propuesta de solución.
- La definición de las tecnología y herramientas que formaron parte del entorno de desarrollo están acorde con los requerimientos actuales del mercado y apuntan a la soberanía tecnológica del país.

Recomendaciones

- Se recomienda realizar pruebas de integración para evaluar el grado de compatibilidad con el sistema del monitor.
- Se recomienda implementar el evaluador de variables empleando tecnologías webviews.

Bibliografía

1. González Landeiro, A., Castillo Galán, D., & Valdés García, L. (2020). MONITOR DE SITIOS WEB: HERRAMIENTA DE APOYO EN LA IMPLEMENTACIÓN DEL GOBIERNO ELECTRÓNICO. Recuperado el 15 de 9 de 2021
2. SEO (Search Engine Optimization): Diario del Economista. Disponible en <https://eleconomista.es/diccionario-de-economia/seo-search-engine-optimization>
3. 40defiebre: Factores que Influyen en el Posicionamiento SEO. Disponible en <https://www.40defiebre.com/guia-seo/como-posiciona-google-una-web>
4. (21 de 11 de 2021). Obtenido de Grill Taylor-Made Performance: <https://digitalgrill.cl/blog-performance-marketing/que-es-google-analytics/>
5. Palacios, A. (23 de 11 de 2021). *Crehana*. Obtenido de <https://www.crehana.com/blog/marketing-digital/que-es-woorank/>
6. *PrensaLink*. (23 de 11 de 2021). Obtenido de https://prensalink.com/woorank/#Que_es_WooRank
7. Nuñez, E. (23 de 11 de 2021). *Crehana*. Obtenido de <https://www.crehana.com/blog/marketing-digital/gtmatrix/>
8. Revuelto, L. (23 de 11 de 2021). Luis Revuelto. Obtenido de <https://www.luisrevuelto.es/tutorial-semrush-2021/>
9. Lenguaje de Programación: Disponible en <https://concepto.de/lenguaje-de-programacion>
10. Wikipedia. Disponible en [https://es.wikipedia.org/wiki/Java_\(lenguaje_de_programaci%C3%B3n\)#En_el_navegador_web](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)#En_el_navegador_web)
11. Lenguaje UML. Disponible en <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>
12. Framework. Disponible en <https://es.m.wikipedia.org/wiki/Framework>
13. Diferencia entre Spring Framework y Spring Boot. Disponible en <https://blog.nubecolectiva.com/cual-es-la-diferencia-entre-spring-framework-y-spring-boot/amp/>
14. NetBeans. Disponible en <https://es.wikipedia.org/wiki/NetBeans>
15. Visual Paradigm. Disponible en <https://modelado10.wordpress.com/2011/11/30/visual-paradigm/>
16. Visual Paradigm. Disponible en <https://mlconsultores.com/visual-paradigm/>
17. MySQL. Disponible en <https://es.m.wikipedia.org/wiki/MySQL>

18. Conozcamos acerca de RabbitMQ, sus componentes y beneficios. Disponible en <https://www.pragma.com.co/academia/lecciones/aprendamos-sobre-rabbitmq-sus-componentes-y-servicios>
19. Acosta, M. P., & Martínez, C. M. de L. (2015). *Sistema informático de apoyo a los procesos “Evaluación del estudiante” y “Censo, capacidad y matrícula” en la Dirección de la Residencia Estudiantil 2 de la Universidad de las Ciencias Informáticas*. Universidad de las Ciencias Informáticas.
20. Saiedian, H. (1997). Una evaluación del del modelo entidad relación extendido. *Information and Software Technology*, 39, 449–462. <https://users.dcc.uchile.cl/~cguierr/cursos/BD/extendido.pdf>
21. Bruzón, F. A. C. (2019). Módulo de gestión de datos de biopsias para el registro de cáncer en el centro Oncológico de Holguín. Universidad de Holguín.
22. Abreu, P. E. A. (2019). *Tema: Mantenimiento al sistema de gestión de pacientes del Centro Oncológico del Hospital General Universitario Vladimir Ilich Lenin*. Universidad de Holguín.
23. EcuRed. (29 de 11 de 2021). Obtenido de https://www.ecured.cu/Diagrama_de_despliegue.
24. *EcuRed*. (22 de 9 de 2010). Recuperado el 29 de 11 de 2021, de https://www.ecured.cu/Patr%C3%B3n_Modelo_Vista_Controlador

Anexos

Tabla 3: HU2-Adaptar los procesos de obtención de contenido

Historia de Usuario	
Número: 2	Nombre: Adaptar los procesos de obtención de contenido
Usuarios: Sistema	
Prioridad en el negocio: Alta	Nivel de complejidad: Alta
Estimación: 4 días	Iteración Asignada: 1
Descripción: Se configura el sistema tomando como base la nueva forma de obtención de contenido	
Observaciones: -	

Tabla 4: HU3-Descargar el contenido HTML de los sitios

Historia de Usuario	
Número: 3	Nombre: Descargar el contenido HTML de los sitios
Usuarios: Sistema	
Prioridad en el negocio: Alta	Nivel de complejidad: Media
Estimación: 4 días	Iteración Asignada: 2
Descripción: El sistema descarga el contenido HTML del sitio web que se va a analizar	
Observaciones: -	

Tabla 5: HU4- Adaptar el sistema a la nueva forma de obtención de contenidos

Historia de Usuario	
Número: 4	Nombre: Adaptar el sistema a la nueva forma de obtención de contenidos
Usuarios: Sistema	
Prioridad en el negocio: Alta	Nivel de complejidad: Alta
Estimación: 4 días	Iteración Asignada: 2
Descripción: El sistema se adapta a la nueva forma de obtención de contenidos, mediante el empleo de tecnologías webviews	
Observaciones: -	

Tabla 6: HU5- Obtener los logs

Historia de Usuario	
Número: 5	Nombre: Obtener los logs
Usuarios: Sistema	
Prioridad en el negocio: Alta	Nivel de complejidad: Media
Estimación: 4 días	Iteración Asignada: 3
Descripción: El sistema obtiene los logs del sitio analizado	
Observaciones: -	

Tabla 7: HU6- Guardar los logs en la base de datos

Historia de Usuario	
Número: 6	Nombre: Guardar los logs en la base de datos
Usuarios: Sistema	
Prioridad en el negocio: Alta	Nivel de complejidad: Media
Estimación: 4 días	Iteración Asignada: 3
Descripción: El sistema guarda los logs recopilados en la base de datos	
Observaciones: -	

Tabla 8: HU7- Integración de Chrome al sistema con webviews

Historia de Usuario	
Número: 7	Nombre: Integración de Chrome al sistema con webviews
Usuarios: Sistema	
Prioridad en el negocio: Alta	Nivel de complejidad: Media
Estimación: 4 días	Iteración Asignada: 3
Descripción: Se integra el Chrome al proceso de descargar de contenido mediante webviews	
Observaciones: -	